PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-271396

(43)Date of publication of application : 20.09.2002

(51)Int.Cl. H04L 12/56
H04L 29/06

(21)Application number : 2001-112676

(22)Date of filing : 11.04.2001

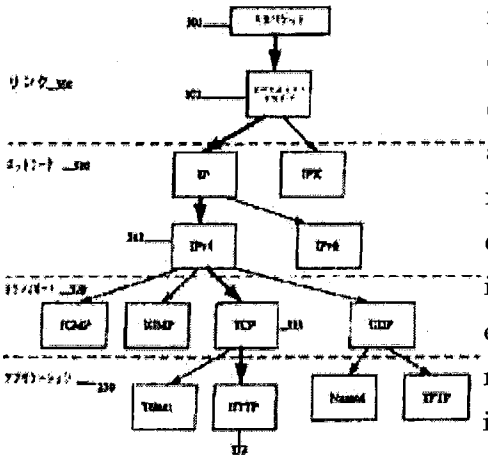
(71)Applicant : INTERNATL BUSINESS MACH CORP <IBM>

(72)Inventor : SCHALES DOUGLAS LEE
SESHAN SRINAVASAN
ZOHAR MIRIAM

(30)Priority

Priority number : 2000 548141 Priority date : 13.04.2000 Priority country : US

(54) NETWORK-DATA-PACKET SORTING AND DEMULTIPLEXING



(57)Abstract:

PROBLEM TO BE SOLVED: To provide a further flexibility in sorting and demultiplexing of packet in a network protocol stack.

SOLUTION: A packet sorting and processing are enlarged by obtaining an external information from the application scheduled outside the scope of kernel transfer or an interrupt context. In one embodiment, the external information may enlarge a reference of node in a sorting tree with an additional information. An enlargement technique for extending the sorting process is provided until the completion of application scheduled outside the scope of kernel transfer or the interrupt context. The resultant external information is used for enlarging the packet sorting. In the other embodiment, the external information may include authorization of sender of the packet by correlating a tunnel ID with a user ID or using s/ident for an authorization of out-of-band. The sort process allows a site policy to practice.

CLAIMS

[Claim(s)]

[Claim 1]It is a method characterized by comprising the following of classifying a data packet, and is a root node of a sorting tree.

A step which receives a data packet.

A step which delivers said data packet to each child of said 1st tree level continuously until the 1st child of the 1st tree level of said sorting tree shows that said 1st child's node standard is satisfied.

A step at which said 1st child forms said data packet in a conformity packet.

A step which repeats delivery and a formation step to said following tree level until the 1st child of the following tree level in the continuing following level stops satisfying a node standard of said 1st child of said following tree level.

[Claim 2]A way according to claim 1 said step to deliver contains a step which performs a code set which returns status directions.

[Claim 3]A way according to claim 1 said step to form contains a step as which said 1st child specifies a code set performed succeedingly.

[Claim 4]A way according to claim 3 said step to specify contains a step which specifies a code set performed following satisfaction.

[Claim 5]A method according to claim 1 containing a step which adds at least 1 node to at least 1 level of said sorting tree dynamically.

[Claim 6]A way according to claim 5 said at least one new child node is the RealAudio node.

[Claim 7]A method characterized by comprising the following of classifying a packet.

A step which postpones an on-going packet classification process to said packet.

A step which acquires external information used in said classification.

[Claim 8]A way according to claim 7 said step to gain contains a step which enlarges a node standard of a node in a sorting tree by said external information.

[Claim 9]A way according to claim 8 said external information includes discernment of dispatch origin of said packet.

[Claim 10]A way according to claim 8 said external information includes attestation of dispatch origin of said packet.

[Claim 11]A way according to claim 7 a classification process is an extensible classification child process.

[Claim 12]A method according to claim 1 of analyzing the syntax of said conformity packet and containing a step which generates pertinent information.

[Claim 13]A method according to claim 1 containing a step which changes said conformity packet into a conversion packet.

[Claim 14]A method according to claim 1 containing a step which relates said packet with the last child [1st] who shows satisfaction.

[Claim 15]A method according to claim 14 of containing a step which performs a code set according to the 1st child of said last.

[Claim 16]A method according to claim 1 containing a step which opts for treatment of said data packet.

[Claim 17]A step which is the method of opting for treatment of a packet received in a child node, and passes the 1st treatment of said packet and this packet to an external process, A way said external process contains a step which enlarges packet treatment, and a step which returns an enlargement packet and enlargement treatment to said child node using a means of process specification.

[Claim 18]A method according to claim 17 containing a step which postpones an on-going treatment process to said packet.

[Claim 19]A way according to claim 18 said enlargement treatment includes discernment of dispatch origin of said packet.

[Claim 20]A way according to claim 18 said enlargement treatment includes attestation of dispatch origin of said packet.

[Claim 21]A method according to claim 18 used for polish enforcement of said treatment.

[Claim 22]A method according to claim 16 containing a step which uses a classification process as a firewall.

[Claim 23]A method according to claim 1 of using a classification process for a classification of an application level.

[Claim 24]A method according to claim 23 of using it for polish enforcement of a classification process.

[Claim 25]A method according to claim 23 of using a classification process for a speed limit.

[Claim 26]A method according to claim 23 of using a classification process for load-balancing-izing.

[Claim 27]A method according to claim 1 of using it for traffic formation of a classification process.

[Claim 28]Are a device which classifies a data packet and a data packet is received from a physical network, A network interface device which passes this data packet to a root node of a sorting tree, and receives a data packet from said root node conversely, and transmits this data packet to said physical network, Until the 1st child node of the next tree level of a sorting tree shows that a node standard of this 1st child node is satisfied, In said following tree level, deliver a packet to a child node continuously from a child node, and until the 1st child node of the continuing following level stops satisfying a node standard of said 1st child node of said continuing following level, A device containing a packet module which forms a data packet in a conformity packet.

[Claim 29]The device according to claim 28 with which said some of devices are realized as an accelerator chip.

[Claim 30]The device according to claim 28 with which said device is used for a classification of an application level.

[Claim 31]The device according to claim 28 with which said device is used as a firewall.

[Claim 32]The device according to claim 28 with which said device is used as a border server.

[Claim 33]A way according to claim 2 said status directions are pm_t types.

[Claim 34]It is a product containing a medium which has a program code means which can be computer read to classify a data packet, and which can be computer read, A

product in which said program code means which can be computer read contains in a computer a program code means which can be computer read to direct to perform a step of Claim 1.

[Claim 35]The product according to claim 34 which said program code means which can be computer read directs adds at least 1 node to at least 1 level of a sorting tree dynamically to a computer.

[Claim 36]It is a product containing a medium which has a program code means which can be computer read to classify a data packet, and which can be computer read, A product in which said program code means which can be computer read contains in a computer a program code means which can be computer read to direct to perform a step of Claim 8.

[Claim 37]It is a product containing a medium which has a program code means which can be computer read to opt for treatment of a packet, and which can be computer read, A product in which said program code means which can be computer read contains in a computer a program code means which can be computer read to direct to perform a step of Claim 18.

[Claim 38]It is a device characterized by comprising the following which classifies a data packet, and is a root node of a sorting tree.

A means to receive a data packet.

A means by which the 1st child node of the 1st tree level of said sorting tree delivers said data packet to each child of said 1st tree level continuously until it shows that a node standard of said 1st child node is satisfied.

A step at which said 1st child forms said data packet in a conformity packet.

A means by which the 1st child node of the following tree level in the continuing following level repeats delivery and a formation step to said following tree level until it stops satisfying a node standard of said 1st child node of said continuing following level.

[Claim 39]A device which opts for treatment of a packet received in a child node, comprising:

An interruption context of a control program in which said child node exists.

An external process besides the range of an interruption context of said control program.

The 1st treatment of said packet and this packet is passed to said external process, A means by which said interruption context receives said enlargement packet and said enlargement treatment from said external process including a means to make said external process enlarge packet treatment by use of a process specifying means, and to make an enlargement packet return to a child node together with enlargement treatment.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention]Especially this invention relates to the classification and demultiplexing of a network communication packet which are processed within a network protocol stack about the field of packet communication.

[0002]

[Description of the Prior Art]In communication through a network, it is required that the information often transported to another computer from a certain computer should be divided into a network communication packet. These network communication packets are only called a "packet", and are transported via a physical communication network.

[0003]By passing various software components, the information emitted from an application program is packet-ized by the network communication packet, is passed to a Network Interface Card after that, and is transmitted on a physical communication network. These software components are hierarchized so that what is known as a network protocol stack may generally be formed. Each class bears responsibility to the facet from which communication differs. For example, a TCP/IP protocol stack is usually divided into four layers, i.e., a link layer, a network layer, the transport layer, and the application layer. Drawing 1 shows the relation between a protocol layer and a TCP/IP protocol stack. The link layer 101 bears the responsibility which arranges data on a physical network. The network layer 102 bears the responsibility of routing, i.e., routing. The transport layer 103 bears the responsibility for communication between two hosts. The application layer 104 bears the responsibility for processing of application specific data.

[0004]For example, drawing 2 shows the stage by which an HTTP request is encapsulated, before being transmitted to a Web server. When a demand descends a protocol stack, each class 201 thru/or 204 encapsulates a packet, and adds the header of itself. If a HTTP packet arrives at a destination address, each protocol layer will classify an ingress packet among all the protocols in the layer of a higher rank rather than it using the information in the header. Generally this process is called demultiplexing (false rumor RUCHIPU REXX).

[0005]In each class in a network protocol stack, demultiplexing of the packet, i.e., "a classification", is carried out from the information in a packet's own data part based on

the information about the packet contained in a header. Based on the classification, packets differ and are processed.

[0006]For example, drawing 3 shows signs that this classification is performed to ingress HTTP request 301. The Ethernet (registered trademark) driver 302 in the link layer 300 classifies a packet based on the frame type in an Ethernet header, and is it IPv4 in the network layer 310 It delivers to 312. IPv4 Based on the IP header protocol value in an IP header, 312 classifies a packet and delivers it to TCP323 in the transport layer 320. Based on the destination port number in a TCP header, TCP323 classifies a packet and delivers it to HTTP server 332 in the application layer 330.

[0007]The conventional packet classification system seen by the firewall of BPF, DPF, Pathfinder, Router Plugins, an operating system, and many is restricted to the set of a fixed pattern-matching rule. This enables a user to monitor or process the arbitrary packets corresponding to the set (usually combine with IP as the protocol header fields, such as a sending agency / destination address, a protocol, or a sending agency / destination port) of the value of the request in a suitable byte range. Next, these packets are passed to a software module, and a software module processes a packet, and is changed, transmitted, removed or delayed in it. Generally, a prominent packet filtering system is based on application traffic, and has the capability to generate and add a rule dynamically. However, such a system does not provide the simple method of extending packet processing so that he may understand a new application protocol.

[0008]A system fully functions to the application which uses the single connection with a well-known destination address and port conventionally [these]. However, for a control session, many newest applications use a well-known service port for the beginning, and then use connection of the addition on a port number for it temporarily for each data stream. The examples of such application are FTP, RealAudio (Real Audio), and H.323. In order to support such applications efficiently, the conventional system must enable dynamic and quick renewal of a packet matching filter rule. Some newest protocols have given up use of a fixed formatted header and the field of fixed size. For example, human being enables it to read a header when HTTP encodes the header as a string.

[0009]

[Problem(s) to be Solved by the Invention]Therefore, the purpose of this invention is to provide bigger pliability in a classification and demultiplexing of the packet in a network protocol stack. As the result, this invention provides the classifying method of an application level. This is based on the below-mentioned classification method and a modular structure.

[0010]Another purpose of this invention is to provide the easy extendibility of packet processing within a network protocol stack by defining the standard method of adding a function or a support new for a new protocol and application.

[0011]Another purpose of this invention is to provide the method and device which acquire external information from the application scheduled out of transmission of a kernel, or the range of an interruption context, in order to enlarge packet sorting or treatment.

[0012]

[Means for Solving the Problem]Working example of this invention is the method of classifying a data packet. This method is provided with the following.

A step which receives a packet in a root node of a sorting tree.

A step which delivers a packet to the 1st child node that suits a node standard of the 1st child node of the 1st tree level of a sorting tree.

A step at which the 1st child node forms a data packet in a conformity packet.

A step which repeats delivery and a formation step to the following tree level until the 1st child node of the next tree level of the continuing following level stops suiting a node standard of the 1st child node of the following tree level.

[0013]In a part of working example, a step to deliver contains a step which performs a code set which returns status directions of a type, A step which shows conformity of a standard performs a code set which identifies a desired packet, Including a step which returns status directions, a step to which a step which forms a data packet in a conformity packet repeats delivery and formation including a step which shows conformity shows incongruent status directions, and contains a step to return.

[0014]In a part of working example of this method, a new child node is the RealAudio node further including a step which adds at least one new child node. Or this method is extensible so that one or more nodes may be dynamically added in an optional label. A step which this method analyzes the syntax of a conformity packet again, and generates pertinent information, A step which changes a conformity packet into a conversion packet, a step which associates a packet in the 1st child node of the last which shows conformity, and a step which performs a code set according to the last child node [1st] are included. Or a step which a step which forms a conformity packet specifies including a step which specifies a code set which the 1st child node continues and is performed specifies a code set performed following a classification.

[0015]Another working example of this invention is the method of classifying a packet using an external process. This method is provided with the following.

A step which postpones an on-going classification process to a packet.

A step which acquires external information used in a classification.
This is performed by application scheduled out of transmission of a kernel, or the range of an interruption context.
[0016]In a part of working example of this method, a step to postpone contains a step which queuing-izes data including information about a packet or its present sentences, and a step which transmits said data to application scheduled out of transmission of a kernel, or the range of an interruption context.
[0017]In a part of working example of this method, a step which acquires external information contains a step which enlarges a node standard of a node in a sorting tree by additional information, A step which a classification process is an extensible classification child process (a process is extensible by adding a new child node in 1 application), and external information specifies including attestation of dispatch origin of a packet includes enforcement of a site polish. A site polish comprises the side in which a large number containing security differ. A security side of a site polish is based on packet sorting and certification information.
[0018]Another mode of this invention is the method of opting for treatment of an original copy packet received in a child node. This method contains a step which delivers treatment of the beginning of an original copy packet and an original copy packet to an external process, A packet which an external process enlarged an original copy packet using a process specifying means, or enlarged the first treatment and was enlarged, and enlarged treatment are returned to a child node. In a part of working example of this method, enlarged treatment includes discernment or attestation of dispatch origin of said packet including a step which postpones an on-going treatment process to an original copy packet.
[0019]
[Embodiment of the Invention]This invention is realizable as a combination of hardware, software or hardware, and software. In execution by the combination of hardware and software, the execution in the computer systems which have a predetermined program is mentioned as a typical example. In this case, by loading this predetermined program to these computer systems, and executing it, this program controls computer systems and performs processing concerning this invention. This program comprises an instruction group which can be expressed by arbitrary language, code, and notations. Such an instruction group makes it possible to perform the function that a system is specific, after conversion for a language, a code, and the notation besides direct or 1, the duplicate to a medium besides 2, **********, or both sides are performed. Of course, this invention contains in the range not only a program such itself but the medium

which recorded the program. The program for performing the function of this invention is storable in the recording medium which can computer read [arbitrary] a floppy (registered trademark) disk, MO, CD-ROM, DVD, a hard disk drive, ROM, MRAM, RAM, etc. This program can be downloaded from other computer systems connected by a communication line for storing in a recording medium, or can be reproduced from other recording media. This program can be compressed, or can be divided into plurality, and can also be stored in a single or multiple recording medium.

[0020]A network protocol is divided into the layer which usually bears responsibility to the facet from which communication differs. For example, drawing 1 shows the network layer of a TCP/IP protocol. The related call graph created by the standard UNIX (registered trademark) protocol stack is constituted like the tree described in relation to drawing 3. Each wooden level corresponds to a different layer in a network protocol stack. This invention copies the call graph of a UNIX protocol stack, and composes a different module which competes a packet in IP layer within a tree structure. Here, a tree structure will be called a sorting tree.

[0021]The example of the sorting tree 400 is shown in drawing 4. Drawing 4 shows each node in a sorting tree as a separate module. Each node is constituted from four packet scanning functions (a matcher, a preprocessor, action, and post processor) and three node management functions (a call-back, a heartbeat, and management) by working example of this invention. Only the packet matching function to identify the packet which should be processed, and the packet action function to opt for packet treatment are required. A packet matching function is called the node standard of a node here. Default specification of the remaining scan and the controlling-function pointer is carried out at NULL. These functions related with each node are memorized by packet filter structure.

[0022]Since each node is a separate module in which loading is dynamically possible, sorting tree composition is supple. In one working example of this invention, a module is loaded to a memory between initialization processes. Based on configuration information, a module is composed and a sorting tree is formed. Modular ordering is important and a packet scan is managed by this ordering. When a sorting tree is created, each node is initialized by performing a code set. In this working example, this code set is a function called a controlling function (mm). The input parameter to mm function is a single pointer which generally points to the buffer containing node specific constitution data.

[0023]Drawing 4 is one example which shows the knitting method of the module in a sorting tree. IPv4 Each of 503, IPv6 504, UDP506, HTTP507, and TCP508 module

wishes the observation or change of a packet which uses the protocol used as the origin of those names. However, in this example, when you wish processing of an HTTP request, two or more methods can be visualized. Use of special TCP for offer of a HTTP proxy function transparent to these methods, and transaction TCP (T/TCP) like HTTP, The execution of filtering of contents based on a site polish or the restriction of packet traffic based on service contract is included. A different module according to the purpose of using a sorting tree is loaded to a memory. A site polish comprises the side in which a large number containing security differ. The security side of a site polish is based on packet sorting and certification information. Completion of initialization may once change a sorting tree by adding, deleting or moving a node. This capability to change a sorting tree makes a packet classification process extensible.

[0024]This invention contains the method of performing a packet classification process and the packet treatment process enlarged. The packet classified or enlarged is called an original copy packet here. The packet of a result is called an enlargement packet. The treatment of an original copy packet is called the 1st treatment here, and the treatment resulting from an enlargement treatment process is called enlargement treatment here. It is claimed that the thing besides transmission of a kernel or the range of an interruption context is external here.

[0025]One working example has seven steps, in order to classify a packet and to opt for enlargement packet treatment. Especially these steps are contained in an interruption context unless it annotates. Steps 1 thru/or 4 show the packet classification process shown in drawing 5. Steps 5 thru/or 7 show enlargement of a packet treatment process. The flow chart of these seven steps is shown in drawing 6. Drawing 5 and drawing 6 are referred to in the following explanation.

[0026]Step 1: A link layer delivers a packet to the root node 502 after receiving a packet from a physical network.

[0027]According to this step, a network driver receives a packet from a physical network, classifies a packet based on the frame type in a MAC header, and delivers to the root node of a sorting tree at it.

[0028]Step 2: A packet is passed to the 1st child node that satisfies the node standard of the child node of the 1st level 521 of a sorting tree.

[0029]A root node asks whether a packet suits the node standard from the left to a child node at the right, and this is continued until the node standard of a child node is satisfied. Next, a root node delivers a packet to the 1st child node that satisfies a node standard, and the 1st child node forms a data packet in a conformity packet. In drawing 5, the root node 502 delivers a packet to the IPv4 node 503 first. The node standard of a

child node contains the code set used in order to identify a desired packet. This code set is realized as a function called the packet matching function (pm) 603.

[0030]The input parameters to pm function are PBUF, the operating system independent data structure containing a packet, an option memory field, and a pointer indicating a packet filter node. The result of a packet matching function shows conformity or the nonconformance of the node standard of a child node, and is a pm_t type. Drawing 7 enumerates the examples of a group of the pm_t type return code value 700. Match_OK, Match_This, Match_Discard, and Match_Forward are contained in the packet matching functional result which shows conformity of the node standard of a child node. The result which shows nonconformance is No_Match.

[0031]A packet matching function like the simple thing which judges whether it agrees in static fixed offset like IPv4 node, FTP and RealAudio, and H.323, There is a complicated thing which identifies the packet for the application which negotiates for additional connection. Though regrettable, since each of such applications has an original method of negotiating for additional connection, an application dependence node is required. This is shown by drawing 8 as H.323 (831), RealAudio 832, and FTP833. A dynamic filter rule is created to connection of each addition. Such dynamic filter rule and other state information about the connection for which it negotiated are locally memorized by the application particular node. In one working example, in order to memorize this data, hash table structure is used. Based on a well-known service port and application specific data, a packet matching function identifies a desired packet and enables the classification of an application level.

[0032]Step 3 : As Step 2 described, it starts from the 1st child node that suits the node standard of the 1st child node of the next tree level of a sorting tree, A "delivery of packet" process is repeated, a packet is formed in a conformity packet, and this processing is continued until neither of the children of the next tree level of a sorting tree stops suiting a node standard (No_Match).

[0033]It is judged whether the next child exists (604). When it exists, a flow is continued to 601. When it does not exist, a flow is continued to 621. After the packet matching function of all the child nodes of the layer of the following tree finishes with an incongruent result, it is said that the packet scanned the sorting tree thoroughly. A scanning path is defined as a node set to the 1st child node of the last which suits the node standard of a child node from a route. In this way, packet sorting is completed and a flow is continued to 621.

[0034]Step 4: In each 1st child node, satisfaction of the node standard of a child node will form a data packet in a conformity packet. This is performed at Steps 4A, 4B, and

4C.

[0035]Step 4A: The present node is added to a node scanning path (605).

[0036]Step 4B: When a code set exists, a node performs a code set, and this analyzes the syntax of and changes a packet (607). A packet is analyzed [the syntax of it] and changed when a code set exists (617). When it does not exist, a flow is continued to 609.

[0037]Once a node standard is satisfied, the scan of the packet of a sorting tree will be restricted to the posterity of a node. The remaining sorting trees are not scanned. However, before scanning the derivation tree of a node, a node may perform a code set. In this example, this code set is called a packet preprocessor function (pp). The input parameter is the same as the case of a packet matching function, and PBUF, the operating system independent data structure containing a packet, an option memory field, and the pointer indicating a packet filter node are contained in them. The return code of pp function is a pp_t type. Drawing 9 enumerates the examples of the pp_t type return code value 900. A packet preprocessor function performs actions, such as syntax analysis of a packet, and conversion of a packet. Information usable for the posterity and ancestor of a node is generated by syntax analysis of a packet. Conversion of a packet is generated when the preprocessor of an IPSec node changes an encryption packet into a decipherment packet, for example. IPSec tunnel information and other information are generated, and these may be used by other nodes in a sorting tree.

[0038]Thus, this invention provides the general-purpose mechanism which saves state information with the mechanism called option transmission (options passing) here, or is transmitted between nodes. In the example of option transmission, an option memory segment is connected to a packet between the tree scans of each packet. Each node uses API, i.e., fw_add_option, and fw_next_option, and memorizes and searches a state. Since the node may be unable to understand all the options passed to it, the option which can understand self is processed and the option which cannot be understood is disregarded.

[0039]Step 4C: By postponing a classification process, a node acquires additional external information and enlarges packet sorting and demultiplexing again.

[0040]Adjournment of a classification process includes arbitrary queuing-izing of data including a packet or the information about the present classification, and the data transfer to the application scheduled out of transmission of a kernel, or the range of an interruption context.

[0041]In one working example, packet sorting is enlarged by postponing a packet classification process until the application scheduled out of transmission of a kernel or the range of an interruption context is completed. The external information of a result is

used in order to enlarge packet sorting.

[0042]A packet discernment agent and a packet certifying agent are contained in the example of the application which enlarges packet sorting. Discernment/certifying agent uses s/ident for discernment out of band and attestation. Attestation uses s/ident for attestation out of band, and is correlation attachment ** to user ID about a packet. At another example of attestation, it is correlation attachment ** to user ID about VPN tunnel ID.

[0043]External information, such as packet discernment or attestation, makes it possible to differ and to process a packet. For example, the site connected to the Internet will assume that bandwidth restriction is carried out strictly. As a result, only a number of employees restricted to arbitrary moments can perform application which has the high bandwidth demand of streaming of data, etc., for example. Based on external information, the site polish which takes preferential treatment against employees' set is realized.

[0044]Step 5: The code set related with the child node of the last which satisfies a node standard is performed after completing packet sorting.

[0045]In one working example, this code set is called a packet action function (pa). Packet action input parameters are PBUF, a pointer indicating a node, a pointer indicating a node scanning path, and an option memory field. The return code of pa function is a paction_t type, and those examples are shown in drawing 10 as 1000. The return code gained opts for packet treatment.

[0046]Usually, the packet action function 621 monitors packet data, and acquires the state information of application specification used by other node functions. For example, the packet action function to have the knowledge of application specification can monitor packet data for the data connection for which it newly negotiates. Such new dynamic connection is locally memorized by the application particular node. A packet matching function uses dynamic data as a part of node standard for the packet sorting of an application level.

[0047]For other examples of the use of a packet action function. Specification change of the packet used for removal of the packet used for queuing-izing of the packet used in order to form change of the packet used in order to realize NAT, and traffic, and a speed limit, and load-balancing-izing is included.

[0048]A packet action function postpones kernel packet processing, and transmits arbitrary data (the information about a packet or its classification is included) to the application scheduled out of transmission of a kernel, or the range of an interruption context again. External information is acquired in order to enlarge packet treatment

(namely, abandonment, transmission, local processing, or specification change) determination (631). Adjournment of a packet treatment determination process includes transmitting data to application using the process specifying means scheduled out of arbitrary queuing-izing of data including the information about a packet or its classification, transmission of a kernel, or the range of an interruption context.

[0049]In one example of the method of enlarging packet treatment determination, a packet treatment determination process on-going [arbitrary] is postponed until the application scheduled out of transmission of a kernel or the range of an interruption context is completed. It is used in order that the external information of a result may enlarge packet treatment determination. The polish enforcement agent and content-filtering agent based on the arbitrary combination of packet sorting, discernment, and attestation are contained in the example of the application which enlarges packet treatment determination. s/identd and an external LDAP server are contained in the example of a process specifying means.

[0050]Once application is completed, application will deliver original data, external information, and a result to a kernel, and a kernel will publish a call to the callback feature of a node. In the node which postponed processing, a callback feature (cb) carries out reinsertion of the packet. A dynamic rule is generated based on the result of application (621).

[0051]For example, especially an advantageous use is concerned with a VPN tunnel. A different polish based on a VPN call destination can enforce using a dynamic rule. These rules are not restricted to fixed pattern matching of a protocol etc. any longer, but are created from the standpoint of application by the classification of an application level. The example of the rule of an application level is "permitting John Doe RealAudio." The rule of an application level simplifies a firewall rule definition in firewall application again.

[0052]Step 6: The code set related with each node after completion of the code set (called a packet action code) related with the child node of the last which satisfies a node standard, and within a node scanning path is performed (623).

[0053]In this example, this code set is called the packet post processor function (px) 625. Packet post-processing input parameters are PBUF, an option memory field, and packet action treatment. The return code of px function is a paction_t type. A paction_t type example is enumerated and shown in drawing 10.

[0054]As packet pretreatment decodes a packet, packet post-processing performs actions, such as encryption of a packet (627). When a packet scans a sorting tree first, a node scanning path is created. node scanning order reverse before returning to a basic

operating system -- the occasion -- packet post-processing is performed.

[0055]Usually, packet treatment is maintained through post-processing. In the case of an abnormal condition, it restricts, and post-processing is not performed after recommendation packet action and pre- post-processing treatment. For example, the outbound tunnel may be destroyed by the VPN tunnel between sorting tree scans.

[0056]Step 7: Control returns to a basic operating system after completion of packet processing, and this processes a packet on abandonment, transmission, specification change, or a partial target based on the last treatment (633).

[0057]Drawing 11 shows working example of this invention as a device which classifies or enlarges the treatment of a data packet. Including the network interface device 1101, this receives a packet from a physical network, and a device passes a packet to the root node of a sorting tree, and receives a packet from a root node conversely, and transmits a packet to a physical network. As for a device, this delivers a packet to a child node continuously from the child node on each tree level including the packet module 1103 further, and this delivery is continued until it shows that the 1st child node of the tree level of a sorting tree satisfies the node standard of that 1st child node. The 1st child node forms a data packet in a conformity packet until the 1st child node of the following level throat in the continuing following level also stops satisfying the node standard of the 1st child node of the following level.

[0058]An accelerator chip may be used in order to realize the packet module 1103. This chip may be used as a classification system of an application level which is needed as the foundation of a firewall box or a border server when diagnosing a high-speed network problem.

[0059]Working example of other devices of this invention may be realized by the person skilled in the art by a known method. For example, this invention is realized using the device which classifies a data packet. In the root node of a sorting tree, the 1st child node of a means to receive a data packet, and the 1st tree level of a sorting tree until this device shows that the node standard of said 1st child node is satisfied, A means by which a data packet is continuously delivered to each child of the 1st tree level, and the 1st child node forms said data packet in a conformity packet, The 1st child node of said next tree level throat of the continuing following level also contains the means which repeats delivery and a formation step to the following tree level until it stops showing that the node standard of said 1st child node of said continuing following level is satisfied. For example, this device takes the gestalt of a floppy disk or a hard disk, a flash memory, or an outside magnetism medium.

[0060]Another working example of this invention is a device which opts for the

treatment of the packet received in a child node. This device is provided with the following.

The interruption context of a control program which a child node interrupts and exists in a context.

The external process besides the range of the interruption context of a control program.

A means for the 1st treatment of said packet and this packet to be delivered to an external process, and for an external process to enlarge packet treatment using a process specifying means, and to return an enlargement packet to a child node together with enlargement treatment.

The interruption context containing a means to receive an enlargement packet and enlargement treatment from an external process.

This device is a gestalt of a hard disk, a floppy disk, or an outside magnetism medium. A control program is realized as software which manages the example of a device.

[0061]This invention is realized by the combination of hardware, software or hardware, and software. It realizes in concentration form within 1 computer systems, or this invention may be realized also by the scatter format in which a different element spreads over two or more computer systems by which interconnection is carried out. It is usable in any kinds of computer systems, or other devices by which adaptation was carried out so that the method described here might be realized. A typical combination of hardware and software, It is a general computer system which has a computer program, and a computer program is loaded, and if it performs, a computer program will control computer systems there and will perform there the method described here. This invention may be embedded in a computer program product again. In this case, including all the features which enable realization of the method described here, a computer program product is loaded to computer systems, and performs these methods.

[0062]Here, a computer program means or a computer program, or [that an instruction set makes the system which has information processing ability perform a specific function directly here by meaning the arbitrary expressions of an instruction set by arbitrary languages, a code, or notation] -- or, 1) Make it perform after either the conversion to another language, a code, or notation and a duplicate with a material gestalt different two and both.

[0063]The above-mentioned explanation describes the purpose of this invention, and some outlines of working example. In the concept of this invention, it is usable to many applications. Therefore, although the above-mentioned explanation describes specific composition and method, the meaning and the concept of this invention are applicable also to other composition and applications. For example, although reference was made

about the data packet, this invention is applicable also like a non-data packet. Probably, it will be clear for other change of working example indicated here to be possible, without swerving from the meaning and the range of this invention, if it is a person skilled in the art. Working example described here therefore, by having only expressed some of the features and applications with which this invention is only prominent, and applying indicated this invention in a different mode, Or by changing this invention into a person skilled in the art by a known method, it will be that other useful advantages are realized. Therefore, it is said again that working example described here is not what was only provided as one example and restricts this invention.

[0064]As a conclusion, the following matters are indicated about the composition of this invention.

[0065](1) In [are the method of classifying a data packet and] the root node of a sorting tree, Until the 1st child of the 1st tree level of said sorting tree indicates it to be a step which receives a data packet to satisfy said 1st child's node standard, The step which delivers said data packet to each child of said 1st tree level continuously, Until the 1st child of the following tree level in the step at which said 1st child forms said data packet in a conformity packet, and the continuing following level stops satisfying the node standard of said 1st child of said following tree level, How to contain the step which repeats delivery and a formation step to said following tree level.

(2) The method of the aforementioned (1) description that said step to deliver contains the step which performs the code set which returns status directions.

(3) The method of the aforementioned (1) description that said step to form contains the step as which said 1st child specifies the code set performed succeedingly.

(4) The method of the aforementioned (3) description that said step to specify contains the step which specifies the code set performed following satisfaction.

(5) The method of the aforementioned (1) description containing the step which adds at least 1 node to at least 1 level of said sorting tree dynamically.

(6) The method of the aforementioned (5) description that said at least one new child node is the RealAudio node.

(7) How to be the method of classifying a packet and contain the step which postpones an on-going packet classification process to said packet, and the step which acquires the external information used in said classification.

(8) The method of the aforementioned (7) description that said step to gain contains the step which enlarges the node standard of the node in a sorting tree by said external information.

(9) The method of the aforementioned (8) description that said external information

includes discernment of the dispatch origin of said packet.

(10) The method of the aforementioned (8) description that said external information includes attestation of the dispatch origin of said packet.

(11) The method of the aforementioned (7) description which is a classification child process with an extensible classification process.

(12) The method of the aforementioned (1) description which analyzes the syntax of said conformity packet and contains the step which generates pertinent information.

(13) The method of the aforementioned (1) description containing the step which changes said conformity packet into a conversion packet.

(14) The method of the aforementioned (1) description containing the step which relates said packet with the last child [1st] who shows satisfaction.

(15) The method of the aforementioned (14) description which contains the step which performs a code set according to the 1st child of said last.

(16) The method of the aforementioned (1) description containing the step which opts for the treatment of said data packet.

(17) The step which is the method of opting for the treatment of the packet received in the child node, and passes the 1st treatment of said packet and this packet to an external process, A way said external process contains the step which enlarges packet treatment, and the step which returns an enlargement packet and enlargement treatment to said child node using the means of process specification.

(18) The method of the aforementioned (17) description containing the step which postpones an on-going treatment process to said packet.

(19) The method of the aforementioned (18) description that said enlargement treatment includes discernment of the dispatch origin of said packet.

(20) The method of the aforementioned (18) description that said enlargement treatment includes attestation of the dispatch origin of said packet.

(21) The method of the aforementioned (18) description used for polish enforcement of said treatment.

(22) The method of the aforementioned (16) description containing the step which uses a classification process as a firewall.

(23) The method of the aforementioned (1) description which uses a classification process for the classification of an application level.

(24) The method of the aforementioned (23) description used for polish enforcement of a classification process.

(25) The method of the aforementioned (23) description which uses a classification process for a speed limit.

(26) The method of the aforementioned (23) description which uses a classification process for load-balancing-izing.
(27) The method of the aforementioned (1) description used for traffic formation of a classification process.
(28) Are a device which classifies a data packet and a data packet is received from a physical network, A network interface device which passes this data packet to the root node of a sorting tree, and receives a data packet from said root node conversely, and transmits this data packet to said physical network, Until the 1st child node of the next tree level of a sorting tree shows that the node standard of this 1st child node is satisfied, In said following tree level, deliver a packet to a child node continuously from a child node, and until the 1st child node of the continuing following level stops satisfying the node standard of said 1st child node of said continuing following level, A device containing the packet module which forms a data packet in a conformity packet.
(29) A device of the aforementioned (28) description with which said some of devices are realized as an accelerator chip.
(30) A device of the aforementioned (28) description with which said device is used for the classification of an application level.
(31) A device of the aforementioned (28) description with which said device is used as a firewall.
(32) A device of the aforementioned (28) description with which said device is used as a border server.
(33) The method of the aforementioned (2) description that said status directions are pm_t types.
(34) It is a product containing the medium which has a program code means which can be computer read to classify a data packet, and which can be computer read, The product in which said program code means which can be computer read contains in a computer a program code means which can be computer read to direct to perform the step of the above (1).
(35) The product of the aforementioned (34) description which said program code means which can be computer read directs adds at least 1 node to at least 1 level of a sorting tree dynamically to a computer.
(36) It is a product containing the medium which has a program code means which can be computer read to classify a data packet, and which can be computer read, The product in which said program code means which can be computer read contains in a computer a program code means which can be computer read to direct to perform the step of the above (8).

(37) It is a product containing the medium which has a program code means which can be computer read to opt for the treatment of a packet, and which can be computer read, The product in which said program code means which can be computer read contains in a computer a program code means which can be computer read to direct to perform the step of the above (18).

(38) In [are a device which classifies a data packet and] the root node of a sorting tree, Until the 1st child node of a means to receive a data packet, and the 1st tree level of said sorting tree shows that the node standard of said 1st child node is satisfied, A means to deliver said data packet to each child of said 1st tree level continuously, Until the 1st child node of the following tree level in the step at which said 1st child forms said data packet in a conformity packet, and the continuing following level stops satisfying the node standard of said 1st child node of said continuing following level, A device which contains the means which repeats delivery and a formation step to said following tree level.

(39) The interruption context of a control program in which it is a device which opts for the treatment of the packet received in a child node, and said child node exists, The external process besides the range of the interruption context of said control program, The 1st treatment of said packet and this packet is passed to said external process, A device which contains a means by which said interruption context receives said enlargement packet and said enlargement treatment from said external process, including a means to make said external process enlarge packet treatment by use of a process specifying means, and to make an enlargement packet return to a child node together with enlargement treatment.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1]It is a figure showing the relation between a protocol layer and a TCP/IP protocol stack.

[Drawing 2]Before being transmitted to a Web server, it is a figure showing the stage by which an HTTP request is encapsulated.

[Drawing 3]It is a figure showing signs that a classification is performed to an ingress HTTP request.

[Drawing 4]It is a figure showing one example of the knitting method of the module in a sorting tree according to this invention.

[Drawing 5]It is a figure showing one example of the packet sorting and the

demultiplexing process of classifying a packet according to this invention.

[Drawing 6]It is a figure showing one example of a step which opts for packet treatment according to this invention.

[Drawing 7]It is a figure showing one according to this invention of pm_t return code.

[Drawing 8]It is a figure showing one example of the application dependence node according to this invention.

[Drawing 9]It is a figure showing one according to this invention of pp_t return code.

[Drawing 10]It is a figure showing one according to this invention of paction_t return code.

[Drawing 11]It is a figure showing one according to this invention of a device.

[Description of Notations]

101, 300 link layers

102, 310 network layers

103, the 320 transport layer

104, the 330 application layers

301 Ingress HTTP request

302 Ethernet driver

312 IPv4

323 TCP

332 HTTP server

502 Root node

503 IPv4

504 IPv6

506 UDP

507 HTTP

508 TCP

521 The 1st level

603 Packet matching function (pm)

621 Packet action function

700 pm_t type return code value

831 H.323

832 RealAudio

833 FTP

900 pp_t type return code value

1101 Network interface device

1103 Packet module

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（A）　(11)特許出願公開番号
特開2002－271396
（P2002－271396A）
(43)公開日　平成14年９月20日(2002.9.20)

(51)Int.Cl.⁷	識別記号	ＦＩ		テーマコード（参考）
Ｈ０４Ｌ　12/56	３００	Ｈ０４Ｌ　12/56	３００Ｄ	５Ｋ０３０
29/06		13/00	３０５Ａ	５Ｋ０３４

審査請求　有　　請求項の数39　ＯＬ　（全 13 頁）

(21)出願番号　特願2001－112676(P2001－112676)

(22)出願日　平成13年４月11日(2001.4.11)

(31)優先権主張番号　０９／５４８１４１

(32)優先日　平成12年４月13日(2000.4.13)

(33)優先権主張国　米国（ＵＳ）

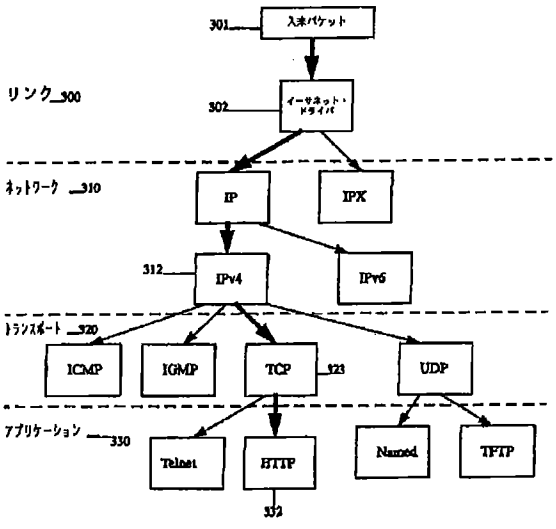
(71)出願人　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
ＩＮＴＥＲＮＡＴＩＯＮＡＬ　ＢＵＳＩＮＥＳＳ　ＭＡＳＣＨＩＮＥＳ　ＣＯＲＰＯＲＡＴＩＯＮ
アメリカ合衆国10504、ニューヨーク州　アーモンク　ニュー　オーチャード　ロード

(74)代理人　100086243
弁理士　坂口　博　（外１名）

最終頁に続く

(54)【発明の名称】　ネットワーク・データ・パケット分類及び逆多重化

(57)【要約】　（修正有）
【課題】　ネットワーク・プロトコル・スタック内のパケットの分類及び逆多重化において、より大きな柔軟性を提供すること。
【解決手段】　カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションから、外部情報を獲得することにより、パケット分類及びパケット処置を増補する。１実施例では、外部情報が追加情報により、分類木内のノードの基準を増補する。カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションが完了するまで、分類プロセスを延期する増補手法が提供される。結果の外部情報は、パケット分類を増補するために使用される。１実施例では、トンネルＩＤをユーザＩＤと相関付けることにより、または帯域外認証のためにs/identを使用することにより、外部情報がパケットの発信元の認証を含む。分類プロセスはサイト・ポリシの施行を可能にする。



1

【特許請求の範囲】

【請求項１】データ・パケットを分類する方法であって、

分類木のルート・ノードにおいて、データ・パケットを受信するステップと、

前記分類木の第１の木レベルの第１の子が、前記第１の子のノード基準を満足することを示すまで、前記第１の木レベルのそれぞれの子に前記データ・パケットを連続的に受け渡すステップと、

前記第１の子が前記データ・パケットを適合パケットに形成するステップと、

続く次のレベルにおける次の木レベルの第１の子が、前記次の木レベルの前記第１の子のノード基準を満足しなくなるまで、前記次の木レベルに対して、受け渡し及び形成ステップを繰り返すステップとを含む、方法。

【請求項２】前記受け渡すステップが、ステータス指示を返却するコード・セットを実行するステップを含む、請求項１記載の方法。

【請求項３】前記形成するステップが、前記第１の子が引き続き実行されるコード・セットを指定するステップを含む、請求項１記載の方法。

【請求項４】前記指定するステップが、満足に続いて実行されるコード・セットを指定するステップを含む、請求項３記載の方法。

【請求項５】前記分類木の少なくとも１レベルに、少なくとも１ノードを動的に追加するステップを含む、請求項１記載の方法。

【請求項６】前記少なくとも１つの新たな子ノードがリアル・オーディオ・ノードである、請求項５記載の方法。

【請求項７】パケットを分類する方法であって、

前記パケットに対して進行中のパケット分類プロセスを延期するステップと、

前記分類において使用される外部情報を獲得するステップとを含む、方法。

【請求項８】前記獲得するステップが、前記外部情報により、分類木内のノードのノード基準を増補するステップを含む、請求項７記載の方法。

【請求項９】前記外部情報が前記パケットの発信元の識別を含む、請求項８記載の方法。

【請求項１０】前記外部情報が前記パケットの発信元の認証を含む、請求項８記載の方法。

【請求項１１】分類プロセスが拡張可能な分類子プロセスである、請求項７記載の方法。

【請求項１２】前記適合パケットを構文解析し、関連情報を生成するステップを含む、請求項１記載の方法。

【請求項１３】前記適合パケットを変換パケットに変換するステップを含む、請求項１記載の方法。

【請求項１４】前記パケットを満足を示す最後の第１の子に関連付けるステップを含む、請求項１記載の方法。

2

【請求項１５】前記最後の第１の子に従い、コード・セットを実行するステップを含む、請求項１４記載の方法。

【請求項１６】前記データ・パケットの処置を決定するステップを含む、請求項１記載の方法。

【請求項１７】子ノードにおいて受信されたパケットの処置を決定する方法であって、

前記パケット及び該パケットの第１の処置を外部プロセスに渡すステップと、

前記外部プロセスがプロセス特定の手段を用いて、パケット処置を増補するステップと、

増補パケット及び増補処置を前記子ノードに戻すステップとを含む、方法。

【請求項１８】前記パケットに対して進行中の処置プロセスを延期するステップを含む、請求項１７記載の方法。

【請求項１９】前記増補処置が前記パケットの発信元の識別を含む、請求項１８記載の方法。

【請求項２０】前記増補処置が前記パケットの発信元の認証を含む、請求項１８記載の方法。

【請求項２１】前記処置がポリシ施行のために使用される、請求項１８記載の方法。

【請求項２２】分類プロセスをファイアウォールとして使用するステップを含む、請求項１６記載の方法。

【請求項２３】分類プロセスをアプリケーション・レベルの分類のために使用する、請求項１記載の方法。

【請求項２４】分類プロセスをポリシ施行のために使用する、請求項２３記載の方法。

【請求項２５】分類プロセスを速度制限のために使用する、請求項２３記載の方法。

【請求項２６】分類プロセスを負荷平衡化のために使用する、請求項２３記載の方法。

【請求項２７】分類プロセスをトラフィック形成のために使用する、請求項１記載の方法。

【請求項２８】データ・パケットを分類する装置であって、

物理ネットワークからデータ・パケットを受信し、該データ・パケットを分類木のルート・ノードに渡し、また逆に、前記ルート・ノードからデータ・パケットを受信し、該データ・パケットを前記物理ネットワークに送信するネットワーク・インタフェース装置と、

分類木の次の木レベルの第１の子ノードが、該第１の子ノードのノード基準を満足することを示すまで、前記次の木レベルにおいて、子ノードから子ノードへとパケットを連続的に受け渡し、続く次のレベルの第１の子ノードが、前記続く次のレベルの前記第１の子ノードのノード基準を満足しなくなるまで、データ・パケットを適合パケットに形成するパケット・モジュールとを含む、装置。

【請求項２９】前記装置の一部がアクセラレータ・チッ

プとして実現される、請求項２８記載の装置。
【請求項３０】前記装置がアプリケーション・レベルの分類に使用される、請求項２８記載の装置。
【請求項３１】前記装置がファイアウォールとして使用される、請求項２８記載の装置。
【請求項３２】前記装置がボーダ・サーバとして使用される、請求項２８記載の装置。
【請求項３３】前記ステータス指示がｐｍ＿ｔタイプである、請求項２記載の方法。
【請求項３４】データ・パケットの分類を行うコンピュータ読取り可能プログラム・コード手段を有するコンピュータ読取り可能媒体を含む製品であって、前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに請求項１のステップを実行するように指示するコンピュータ読取り可能プログラム・コード手段を含む製品。
【請求項３５】前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに分類木の少なくとも１レベルに少なくとも１ノードを動的に追加するように指示する、請求項３４記載の製品。
【請求項３６】データ・パケットの分類を行うコンピュータ読取り可能プログラム・コード手段を有するコンピュータ読取り可能媒体を含む製品であって、前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに請求項８のステップを実行するように指示するコンピュータ読取り可能プログラム・コード手段を含む製品。
【請求項３７】パケットの処置の決定を行うコンピュータ読取り可能プログラム・コード手段を有するコンピュータ読取り可能媒体を含む製品であって、前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに請求項１８のステップを実行するように指示するコンピュータ読取り可能プログラム・コード手段を含む製品。
【請求項３８】データ・パケットを分類する装置であって、
分類木のルート・ノードにおいて、データ・パケットを受信する手段と、
前記分類木の第１の木レベルの第１の子ノードが、前記第１の子ノードのノード基準を満足することを示すまで、前記第１の木レベルのそれぞれの子に前記データ・パケットを連続的に受け渡す手段と、
前記第１の子が前記データ・パケットを適合パケットに形成するステップと、
続く次のレベルにおける次の木レベルの第１の子ノードが、前記続く次のレベルの前記第１の子ノードのノード基準を満足しなくなるまで、前記次の木レベルに対して、受け渡し及び形成ステップを繰り返す手段とを含む、装置。
【請求項３９】子ノードにおいて受信されるパケットの処置を決定する装置であって、
前記子ノードが存在する、制御プログラムの割込みコンテキストと、
前記制御プログラムの割込みコンテキストの範囲外の外部プロセスと、
前記パケット及び該パケットの第１の処置を前記外部プロセスに渡し、前記外部プロセスにプロセス特定手段の使用によりパケット処置を増補させ、増補パケットを増補処置と一緒に子ノードに返却させる手段とを含み、前記割込みコンテキストが、前記増補パケット及び前記増補処置を前記外部プロセスから受信する手段を含む、装置。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明はパケット通信の分野に関し、特に、ネットワーク・プロトコル・スタック内で処理されるネットワーク通信パケットの分類及び逆多重化に関する。
【０００２】
【従来の技術】ネットワークを介する通信では、しばしば、あるコンピュータから別のコンピュータに移送される情報が、ネットワーク通信パケットに分割されることが要求される。これらのネットワーク通信パケットは、単に"パケット"と称され、物理通信ネットワークを介して移送される。
【０００３】アプリケーション・プログラムから発する情報は、様々なソフトウェア・コンポーネントを通過することにより、ネットワーク通信パケットにパケット化され、その後、ネットワーク・インタフェース・カードに渡されて、物理通信ネットワーク上に伝送される。これらのソフトウェア・コンポーネントは、一般に、ネットワーク・プロトコル・スタックとして知られるものを形成するように階層化される。各層は、通信の異なるファセットに対して責任を担う。例えば、ＴＣＰ／ＩＰプロトコル・スタックは通常、４つの層、すなわちリンク層、ネットワーク層、トランスポート層、及びアプリケーション層に分割される。図１は、プロトコル層とＴＣＰ／ＩＰプロトコル・スタックとの関係を示す。リンク層１０１は、データを物理ネットワーク上に配置する責任を担う。ネットワーク層１０２は、ルーティングすなわち経路指定の責任を担う。トランスポート層１０３は、２つのホスト間の通信の責任を担う。アプリケーション層１０４は、アプリケーション特定データの処理の責任を担う。
【０００４】例えば、図２は、ウェブ・サーバに送信される前に、ＨＴＴＰ要求がカプセル化されるステージを示す。要求がプロトコル・スタックを降下するとき、各層２０１乃至２０４がパケットをカプセル化し、それ自身のヘッダを追加する。ＨＴＴＰパケットが宛先アドレスに到着すると、各プロトコル層がそのヘッダ内の情報

を用いて、入来パケットをそれよりも上位の層内の全てのプロトコルの間で分類する。このプロセスは一般に、逆多重化（デマルチプレックス）と称される。

【０００５】ネットワーク・プロトコル・スタック内の各層において、ヘッダ内に含まれるパケットに関する情報にもとづき、またはパケット自身のデータ部分内の情報からパケットが逆多重化、すなわち"分類"される。パケットはその分類にもとづき、異なって処理される。

【０００６】例えば、図３は、この分類が入来ＨＴＴＰ要求３０１に対して実行される様子を示す。リンク層３００内のイーサネット（登録商標）・ドライバ３０２は、イーサネット・ヘッダ内のフレーム・タイプにもとづきパケットを分類し、それをネットワーク層３１０内のＩＰｖ４　３１２に受け渡す。ＩＰｖ４　３１２は、ＩＰヘッダ内のＩＰヘッダ・プロトコル値にもとづき、パケットを分類し、それをトランスポート層３２０内のＴＣＰ３２３に受け渡す。ＴＣＰ３２３は、ＴＣＰヘッダ内の宛先ポート番号にもとづき、パケットを分類し、それをアプリケーション層３３０内のＨＴＴＰサーバ３３２に受け渡す。

【０００７】ＢＰＦ、ＤＰＦ、Pathfinder、Router Plugins、オペレーティング・システム及び多くのファイアウォールで見られる従来のパケット分類システムは、固定のパターン・マッチング規則のセットに制限される。これはユーザが、適当なバイト範囲内の所望の値のセット（通常、ＩＰと、発信元／宛先アドレス、プロトコルまたは発信元／宛先ポートなどのプロトコル・ヘッダ・フィールドとの組み合わせ）に合致する任意のパケットを傍受または処理することを可能にする。次に、これらのパケットがソフトウェア・モジュールに渡され、ソフトウェア・モジュールがパケットを処理し、それを変更、転送、除去または遅延したりする。著名なパケット・フィルタリング・システムは一般に、アプリケーション・トラフィックにもとづき、規則を動的に生成及び追加する能力を有する。しかしながら、こうしたシステムは、新たなアプリケーション・プロトコルを理解するように、パケット処理を拡張する単純な方法を提供しない。

【０００８】これらの従来システムは、周知の宛先アドレス及びポートへの単一の接続を使用するアプリケーションに対しては十分に機能する。しかしながら、多くの最新のアプリケーションは、最初に制御セッションのために周知のサービス・ポートを使用し、次に各データ・ストリームのために、一時ポート番号上での追加の接続を使用する。こうしたアプリケーションの例は、ＦＴＰ、リアル・オーディオ（Real Audio）、及びH．３２３である。これらのアプリケーションを効率的にサポートするために、従来のシステムはパケット・マッチング・フィルタ規則の動的且つ迅速な更新を可能にしなければならない。更に、一部の最新のプロトコルは、固定のフォーマット・ヘッダ及び固定サイズのフィールドの使用を断念している。例えば、ＨＴＴＰは、そのヘッダをストリングとしてエンコードすることにより人間がヘッダを読めるようにする。

【０００９】

【発明が解決しようとする課題】従って、本発明の目的は、ネットワーク・プロトコル・スタック内のパケットの分類及び逆多重化において、より大きな柔軟性を提供することである。その結果として、本発明はアプリケーション・レベルの分類方法を提供する。これは後述の分類手法、及びモジュラ構造による。

【００１０】本発明の別の目的は、新たなプロトコル及びアプリケーションのために、新たな機能またはサポートを追加する標準的な方法を定義することにより、ネットワーク・プロトコル・スタック内での、パケット処理の容易な拡張性を提供することである。

【００１１】更に本発明の別の目的は、パケット分類または処置を増補するために、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションから、外部情報を獲得する方法及び装置を提供することである。

【００１２】

【課題を解決するための手段】本発明の実施例は、データ・パケットを分類する方法である。この方法は、分類木のルート・ノードにおいて、パケットを受信するステップと、パケットを分類木の第１の木レベルの第１の子ノードのノード基準に適合する、第１の子ノードに受け渡すステップと、第１の子ノードがデータ・パケットを適合パケットに形成するステップと、続く次のレベルの次の木レベルの第１の子ノードが、次の木レベルの第１の子ノードのノード基準に適合しなくなるまで、次の木レベルに対して、受け渡し及び形成ステップを繰り返すステップとを含む。

【００１３】一部の実施例では、受け渡すステップが、タイプのステータス指示を返却するコード・セットを実行するステップを含み、基準の適合を示すステップが、所望のパケットを識別するコード・セットを実行し、ステータス指示を返却するステップを含み、データ・パケットを適合パケットに形成するステップが、適合を示すステップを含み、受け渡し及び形成を繰り返すステップが、不適合のステータス指示を示し、返却するステップを含む。

【００１４】本方法の一部の実施例では、更に、少なくとも１つの新たな子ノードを追加するステップを含み、新たな子ノードがリアル・オーディオ・ノードである。或いは、本方法は、１つ以上のノードが任意レベルにおいて動的に追加されるように拡張可能である。本方法はまた、適合パケットを構文解析し、関連情報を生成するステップと、適合パケットを変換パケットに変換するステップと、適合を示す最後の第１の子ノードにおいてパ

ケットを関連付けるステップと、最後の第１の子ノードに従い、コード・セットを実行するステップとを含む。或いは、適合パケットを形成するステップが、第１の子ノードが続いて実行されるコード・セットを指定するステップを含み、指定するステップが分類に続き実行されるコード・セットを指定する。

【００１５】本発明の別の実施例は、外部プロセスを用いてパケットを分類する方法である。この方法は、パケットに対して進行中の分類プロセスを延期するステップと、分類において使用される外部情報を獲得するステップとを含む。これはカーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションにより実行される。

【００１６】本方法の一部の実施例では、延期するステップが、パケットまたはその現文類に関する情報を含むデータを待ち行列化するステップと、前記データを、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションに転送するステップとを含む。

【００１７】本方法の一部の実施例では、外部情報を獲得するステップが、分類木内のノードのノード基準を追加情報により増補するステップを含み、外部情報がパケットの発信元の認証を含み、分類プロセスが拡張可能な分類子プロセスであり（１アプリケーションでは、新たな子ノードを追加することにより、プロセスが拡張可能）、指定するステップがサイト・ポリシの施行を含む。サイト・ポリシは、セキュリティを含む多数の異なる側面から構成される。サイト・ポリシのセキュリティ面は、パケット分類及び認証情報にもとづいたりする。

【００１８】本発明の別の態様は、子ノードにおいて受信されるオリジナル・パケットの処置を決定する方法である。この方法は、オリジナル・パケット及びオリジナル・パケットの最初の処置を外部プロセスに受け渡すステップを含み、外部プロセスがプロセス特定手段を用いて、オリジナル・パケットを増補するか、最初の処置を増補し、増補されたパケット及び増補された処置を子ノードに返却する。この方法の一部の実施例では、オリジナル・パケットに対して進行中の処置プロセスを延期するステップを含み、増補された処置が、前記パケットの発信元の識別または認証を含む。

【００１９】

【発明の実施の形態】本発明は、ハードウエア、ソフトウエア、またはハードウエア及びソフトウエアの組み合わせとして実現可能である。ハードウエアとソフトウエアの組み合わせによる実行において、所定のプログラムを有するコンピュータ・システムにおける実行が典型的な例として挙げられる。かかる場合、該所定プログラムが該コンピュータ・システムにロードされ実行されることにより、該プログラムは、コンピュータ・システムを制御し、本発明にかかる処理を実行させる。このプログラムは、任意の言語・コード・表記によって表現可能な命令群から構成される。そのような命令群は、システムが特定の機能を直接、または１）他の言語・コード・表記への変換、２）他の媒体への複製、のいずれか一方もしくは双方が行われた後に、実行する事を可能にするものである。もちろん、本発明は、そのようなプログラム自体のみならず、プログラムを記録した媒体もその範囲に含むものである。本発明の機能を実行するためのプログラムは、フロッピー（登録商標）・ディスク、ＭＯ、ＣＤ－ＲＯＭ、ＤＶＤ、ハード・ディスク装置、ＲＯＭ、ＭＲＡＭ、ＲＡＭ等の任意のコンピュータ読取り可能な記録媒体に格納することができる。かかるプログラムは、記録媒体への格納のために、通信回線で接続する他のコンピュータ・システムからダウンロードしたり、他の記録媒体から複製したりすることができる。また、かかるプログラムは、圧縮し、または複数に分割して、単一または複数の記録媒体に格納することもできる。

【００２０】ネットワーク・プロトコルは通常、通信の異なるファセットに対して責任を担う層に分割される。例えば、図１はＴＣＰ／ＩＰプロトコルのネットワーク層を示す。標準のＵＮＩＸ（登録商標）プロトコル・スタックにより作成される関連コール・グラフは、図３に関連して述べる木のように構成される。木の各レベルは、ネットワーク・プロトコル・スタック内の異なる層に対応する。本発明はＵＮＩＸプロトコル・スタックのコール・グラフを模倣し、木構造内のＩＰ層においてパケットを競合する異なるモジュールを編成する。尚、ここでは木構造を分類木と呼ぶことにする。

【００２１】分類木４００の例が、図４に示される。図４は、分類木内の各ノードを別々のモジュールとして示す。本発明の実施例では、各ノードが４つのパケット走査機能（マッチャ、プリプロセッサ、アクション、及びポスト・プロセッサ）と、３つのノード管理機能（コールバック、ハートビート、及び管理）とから構成される。処理すべきパケットを識別するパケット・マッチング機能と、パケット処置を決定するパケット・アクション機能だけが要求される。パケット・マッチング機能は、ここではノードのノード基準と称される。残りの走査及び管理機能ポインタは、NULLにデフォルト指定される。各ノードに関連付けられるこれらの機能は、パケット・フィルタ構造に記憶される。

【００２２】各ノードは別々の動的にロード可能なモジュールであるので、分類木構成は柔軟性がある。本発明の１実施例では、初期化プロセスの間に、モジュールがメモリにロードされる。構成情報にもとづき、モジュールが編成されて、分類木を形成する。モジュールの順序付けは重要であり、この順序付けによりパケット走査が管理される。分類木が作成されるとき、コード・セットを実行することにより各ノードが初期化される。この実施例では、このコード・セットは管理機能（ｍｍ）と称

される機能である。mm機能への入力パラメータは、一般に、ノード特定構成データを含むバッファを指し示す単一のポインタである。

【００２３】図４は、分類木内のモジュールの編成方法を示す１例である。ＩＰｖ４　５０３、ＩＰｖ６　５０４、ＵＤＰ５０６、ＨＴＴＰ５０７、及びＴＣＰ５０８モジュールの各々は、それらの名前の由来となるプロトコルを使用するパケットの観察または変更を希望する。しかしながら、この例では、ＨＴＴＰ要求の処理を希望するとき、複数の方法を思い描くことができる。これらの方法には、透過的なＨＴＴＰプロキシ機能の提供、ＨＴＴＰのようなトランザクションＴＣＰ（Ｔ／ＴＣＰ）のための特殊なＴＣＰの使用、サイト・ポリシにもとづくコンテンツのフィルタリングの実行、或いはサービス契約にもとづくパケット・トラフィックの制限が含まれる。分類木の使用目的に応じて異なるモジュールがメモリにロードされる。サイト・ポリシは、セキュリティを含む多数の異なる側面から構成される。サイト・ポリシのセキュリティ面は、パケット分類及び認証情報にもとづいたりする。一旦、初期化が完了するとノードを追加、削除、または移動することにより、分類木が変更され得る。分類木を変更するこの能力は、パケット分類プロセスを拡張可能にする。

【００２４】本発明は、パケット分類プロセス及び増補されるパケット処置プロセスを実行する方法を含む。分類または増補されるパケットは、ここではオリジナル・パケットと称される。結果のパケットは、増補パケットと称される。オリジナル・パケットの処置は、ここでは第１の処置と称され、増補処置プロセスに起因する処置は、ここでは増補処置と称される。カーネルの転送または割込みコンテキストの範囲外のものは、ここでは外部的と称される。

【００２５】１実施例は、パケットを分類し、増補パケット処置を決定するために、７ステップを有する。これらのステップは、特に注記しない限り、割込みコンテキストに含まれる。ステップ１乃至４は、図５に示されるパケット分類プロセスを示す。ステップ５乃至７は、パケット処置プロセスの増補を示す。これらの７ステップのフロー図が、図６に示される。以下の説明では、図５及び図６を参照する。

【００２６】ステップ１：物理ネットワークからパケットを受信後、リンク層がパケットをルート・ノード５０２に受け渡す。

【００２７】このステップでは、ネットワーク・ドライバがパケットを物理ネットワークから受信し、パケットをＭＡＣヘッダ内のフレーム・タイプにもとづいて分類し、分類木のルート・ノードに受け渡す。

【００２８】ステップ２：パケットが、分類木の第１レベル５２１の子ノードのノード基準を満足する第１の子ノードに渡される。

【００２９】ルート・ノードは、子ノードに対して左から右に、パケットがそのノード基準に適合するか否かを問い合わせ、これは子ノードのノード基準が満足されるまで継続される。次にルート・ノードがパケットを、ノード基準を満足するその第１の子ノードに受け渡し、第１の子ノードがデータ・パケットを適合パケットに形成する。図５では、ルート・ノード５０２が最初にパケットをＩＰｖ４ノード５０３に受け渡す。子ノードのノード基準は、所望のパケットを識別するために使用されるコード・セットを含む。このコード・セットは、パケット・マッチング機能（ｐｍ）６０３と呼ばれる機能として実現される。

【００３０】ｐｍ機能への入力パラメータは、ＰＢＵＦ、パケットを含むオペレーティング・システム独立データ構造、オプション・メモリ領域、及びパケット・フィルタ・ノードを指し示すポインタである。パケット・マッチング機能の結果は、子ノードのノード基準の適合または不適合を示し、ｐｍ__ｔタイプである。図７は、ｐｍ__ｔタイプ戻りコード値７００のグループ例を列挙する。子ノードのノード基準の適合を示すパケット・マッチング機能結果には、Match_OK、Match_This、Match_Discard、及びMatch_Forwardが含まれる。不適合を示す結果は、No_Matchである。

【００３１】パケット・マッチング機能は、ＩＰｖ４ノードなどのように、静的な固定オフセットに合致するか否かを判断する単純なものと、ＦＴＰ、リアル・オーディオ、及びH．３２３などのように、追加の接続を折衝するアプリケーションのためのパケットを識別する複雑なものとがある。残念ながら、これらのアプリケーションの各々は、追加の接続を折衝する独自の方法を有するので、アプリケーション依存ノードが要求される。これは図８では、H．３２３（８３１）、リアル・オーディオ８３２、及びＦＴＰ８３３として示される。各追加の接続に対して動的フィルタ規則が作成される。折衝された接続に関する、これらの動的フィルタ規則及び他の状態情報は、アプリケーション特定ノードに局所的に記憶される。１実施例では、このデータを記憶するためにハッシュ・テーブル構造を使用する。周知のサービス・ポート及びアプリケーション特定データにもとづき、パケット・マッチング機能は所望のパケットを識別し、アプリケーション・レベルの分類を可能にする。

【００３２】ステップ３：ステップ２で述べたように、分類木の次の木レベルの第１の子ノードのノード基準に適合する第１の子ノードから開始して、"パケットの受け渡し"プロセスを繰り返し、パケットを適合パケットに形成し、この処理を分類木の次の木レベルのいずれの子もノード基準に適合しなくなるまで（No_Match）継続する。

【００３３】次の子が存在するか否かが判断される（６０４）。存在する場合、フローは６０１に継続する。存

在しない場合、フローは６２１に継続する。次の木の層の全ての子ノードのパケット・マッチング機能が不適合結果に終わると、パケットは完全に分類木を走査したと言われる。走査経路は、ルートから子ノードのノード基準に適合する最後の第１の子ノードへのノード・セットとして定義される。こうしてパケット分類が完了し、フローは６２１に継続する。

【００３４】ステップ４：各第１の子ノードにおいて、子ノードのノード基準が満足されると、データ・パケットを適合パケットに形成する。これはステップ４Ａ、４Ｂ及び４Ｃで実行される。

【００３５】ステップ４Ａ：現ノードがノード走査経路に追加される（６０５）。

【００３６】ステップ４Ｂ：コード・セットが存在する場合、ノードがコード・セットを実行し、これがパケットを構文解析し、変換する（６０７）。コード・セットが存在する場合、パケットが構文解析され、変換される（６１７）。存在しない場合、フローは６０９に継続する。

【００３７】一旦ノード基準が満足されると、分類木のパケットの走査はノードの子孫に制限される。残りの分類木は走査されない。しかしながら、ノードの派生木を走査する前に、ノードはコード・セットを実行してもよい。本実施例では、このコード・セットはパケット・プリプロセッサ機能（ｐｐ）と称される。入力パラメータはパケット・マッチング機能の場合と同じであり、それらにはＰＢＵＦ、パケットを含むオペレーティング・システム独立データ構造、オプション・メモリ領域、及びパケット・フィルタ・ノードを指し示すポインタが含まれる。ｐｐ機能の戻りコードは、ｐｐ＿ｔタイプである。図９は、ｐｐ＿ｔタイプ戻りコード値９００の例を列挙する。パケット・プリプロセッサ機能は、パケットの構文解析やパケットの変換などのアクションを実行する。パケットの構文解析により、ノードの子孫及び先祖にとって使用可能な情報が生成される。パケットの変換は、例えばＩＰＳｅｃノードのプリプロセッサが暗号化パケットを解読パケットに変換するとき、発生する。ＩＰＳｅｃトンネル情報及び他の情報が生成され、これらが分類木内の他のノードにより使用され得る。

【００３８】このように、本発明はここではオプション転送（options passing）と呼ぶ機構により状態情報を保存したり、ノード間で転送する汎用機構を提供する。オプション転送の例では、各パケットの木走査の間にオプション・メモリ・セグメントがパケットに接続される。各ノードはＡＰＩ、すなわちfw_add_option及びfw_next_optionを用いて状態を記憶及び検索する。ノードはそれに渡される全てのオプションを理解できない可能性があるので、自身が理解できるオプションを処理し、理解できないオプションは無視する。

【００３９】ステップ４Ｃ：ノードはまた、分類プロセスを延期することにより、追加の外部情報を獲得し、パケット分類及び逆多重化を増補する。

【００４０】分類プロセスの延期は、パケットまたはその現分類に関する情報を含む任意のデータの待ち行列化と、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションへのデータの転送を含む。

【００４１】１実施例では、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションが完了するまで、パケット分類プロセスを延期することによりパケット分類を増補する。結果の外部情報が、パケット分類を増補するために使用される。

【００４２】パケット分類を増補するアプリケーションの例には、パケット識別エージェント及びパケット認証エージェントが含まれる。識別／認証エージェントは、帯域外識別及び認証のために、s/identを使用する。認証は帯域外認証のためにs/identを使用し、パケットをユーザＩＤに相関付ける。認証の別の例では、ＶＰＮトンネルＩＤをユーザＩＤに相関付ける。

【００４３】パケット識別または認証などの外部情報は、パケットを異なって処理することを可能にする。例えば、インターネットに接続されるサイトが、厳格に帯域幅制限されていると仮定しよう。その結果、任意の瞬間に限られた数の従業員だけが、例えばデータのストリーミングなどの高い帯域幅要求を有するアプリケーションを実行できる。外部情報にもとづき、従業員の集合に対して優遇措置を講じるサイト・ポリシが実現される。

【００４４】ステップ５：パケット分類が完了後、ノード基準を満足する最後の子ノードに関連付けられるコード・セットが実行される。

【００４５】１実施例では、このコード・セットはパケット・アクション機能（ｐａ）と称される。パケット・アクション入力パラメータは、ＰＢＵＦ、ノードを指し示すポインタ、ノード走査経路を指し示すポインタ、及びオプション・メモリ領域である。ｐａ機能の戻りコードはpaction_tタイプであり、それらの例が図１０に１０００として示される。獲得される戻りコードがパケット処置を決定する。

【００４６】通常、パケット・アクション機能６２１はパケット・データをモニタし、他のノード機能により使用されるアプリケーション特定の状態情報を獲得する。例えば、アプリケーション特定の知識を有するパケット・アクション機能は、新たに折衝されるデータ接続のために、パケット・データをモニタすることができる。これらの新たな動的接続は、アプリケーション特定ノードに局所的に記憶される。パケット・マッチング機能は動的データを、アプリケーション・レベルのパケット分類のためのノード基準の一部として使用する。

【００４７】パケット・アクション機能の用途の他の例には、ＮＡＴを実現するために使用されるパケットの変

更、トラフィックを形作るために使用されるパケットの待ち行列化、速度制限のために使用されるパケットの除去、及び負荷平衡化のために使用されるパケットの指定変更が含まれる。

【００４８】パケット・アクション機能はまた、カーネル・パケット処理を延期し、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションに、任意のデータ（パケットまたはその分類に関する情報を含む）を転送する。また、パケット処置（すなわち廃棄、転送、局所的処理または指定変更）決定（６３１）を増補するために外部情報を獲得する。パケット処置決定プロセスの延期は、パケットまたはその分類に関する情報を含む任意のデータの待ち行列化と、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるプロセス特定手段を用いて、データをアプリケーションに転送することとを含む。

【００４９】パケット処置決定を増補する方法の１例では、カーネルの転送または割込みコンテキストの範囲外でスケジュールされるアプリケーションが完了するまで、任意の進行中のパケット処置決定プロセスを延期する。結果の外部情報がパケット処置決定を増補するために使用される。パケット処置決定を増補するアプリケーションの例には、パケット分類、識別及び認証の任意の組み合わせにもとづく、ポリシ施行エージェント及びコンテンツ・フィルタリング・エージェントが含まれる。プロセス特定手段の例には、s/identd及び外部ＬＤＡＰサーバが含まれる。

【００５０】一旦アプリケーションが完了すると、アプリケーションはオリジナル・データ、外部情報及び結果をカーネルに受け渡し、カーネルがノードのコールバック機能に呼び出しを発行する。コールバック機能（ｃｂ）は、処理を延期したノードにおいてパケットを再挿入する。アプリケーションの結果にもとづき、動的規則が生成される（６２１）。

【００５１】例えば特に有利な用途は、ＶＰＮトンネルに関わる。ＶＰＮ呼び出し先にもとづく異なるポリシが、動的規則を用いて施行可能である。アプリケーション・レベルの分類により、これらの規則はもはやプロトコルなどの固定パターン・マッチングに制限されず、アプリケーションの見地から作成される。アプリケーション・レベルの規則の例は、"John Doeにリアル・オーディオを許可する"である。アプリケーション・レベルの規則はまた、ファイアウォール・アプリケーションにおいて、ファイアウォール規則定義を単純化する。

【００５２】ステップ６：ノード基準を満足する最後の子ノードに関連付けられるコード・セット（パケット・アクション・コードと称される）の完了後、ノード走査経路内の各ノードに関連付けられるコード・セットが実行される（６２３）。

【００５３】本実施例では、このコード・セットはパケット・ポスト・プロセッサ機能（ｐｘ）６２５と称される。パケット後処理入力パラメータは、ＰＢＵＦ、オプション・メモリ領域、及びパケット・アクション処置である。ｐｘ機能の戻りコードは、paction_tタイプである。paction_tタイプの例が、図１０に列挙されて示される。

【００５４】パケット前処理がパケットを解読するのと同様、パケット後処理は、パケットの暗号化などのアクションを実行する（６２７）。パケットが最初に分類木を走査するとき、ノード走査経路が作成される。基本オペレーティング・システムに戻る前に、逆のノード走査順序でパケット後処理が実行される。

【００５５】通常、パケット処置は後処理を通じて保たれる。異常状態の場合に限り、後処理は推奨パケット・アクション及び前の後処理処置の後に実行されない。例えば、ＶＰＮトンネルにより、アウトバウンド・トンネルが分類木走査の間に破壊されてしまっているかもしれない。

【００５６】ステップ７：パケット処理の完了後、制御は基本オペレーティング・システムに戻り、これが最終処置にもとづき、パケットを廃棄、転送、指定変更、または局所的に処理する（６３３）。

【００５７】図１１は、データ・パケットの処置を分類または増補する装置としての本発明の実施例を示す。装置はネットワーク・インタフェース装置１１０１を含み、これは物理ネットワークからパケットを受信し、パケットを分類木のルート・ノードに渡し、また逆に、ルート・ノードからパケットを受信し、パケットを物理ネットワークに送信する。装置は更にパケット・モジュール１１０３を含み、これは各木レベル上の子ノードから子ノードへとパケットを連続的に受け渡し、この受け渡しは分類木の木レベルの第１の子ノードが、その第１の子ノードのノード基準を満足することを示すまで継続される。第１の子ノードは、続く次のレベルにおける次のレベルのどの第１の子ノードも、次のレベルの第１の子ノードのノード基準を満足しなくなるまで、データ・パケットを適合パケットに形成する。

【００５８】パケット・モジュール１１０３を実現するために、アクセラレータ・チップが使用され得る。このチップは、ファイアウォール・ボックスやボーダ・サーバの基礎として、或いは、高速ネットワーク問題を診断する場合に必要とされるような、アプリケーション・レベルの分類システムとして使用され得る。

【００５９】本発明の他の装置の実施例は、当業者には既知の方法で実現され得る。例えば、本発明はデータ・パケットを分類する装置を用いて実現される。この装置は、分類木のルート・ノードにおいて、データ・パケットを受信する手段と、分類木の第１の木レベルの第１の子ノードが、前記第１の子ノードのノード基準を満足することを示すまで、データ・パケットを第１の木レベル

のそれぞれの子に連続的に受け渡し、第１の子ノードが前記データ・パケットを適合パケットに形成する手段と、続く次のレベルの前記次の木レベルのどの第１の子ノードも、前記続く次のレベルの前記第１の子ノードのノード基準を満足することを示さなくなるまで、次の木レベルに対して、受け渡し及び形成ステップを繰り返す手段とを含む。例えば、この装置は、フロッピー・ディスクまたはハード・ディスク、フラッシュ・メモリ、または外部磁気媒体などの形態を取る。

【００６０】本発明の別の実施例は、子ノードにおいて受信されるパケットの処置を決定する装置である。この装置は、子ノードが割込みコンテキスト内に存在する、制御プログラムの割込みコンテキストと、制御プログラムの割込みコンテキストの範囲外の外部プロセスと、前記パケット及び該パケットの第１の処置を外部プロセスに受け渡し、外部プロセスがプロセス特定手段を用いて、パケット処置を増補し、増補パケットを増補処置と一緒に子ノードに返却する手段と、増補パケット及び増補処置を外部プロセスから受信する手段を含む割込みコンテキストとを含む。この装置は例えば、ハード・ディスク、フロッピー・ディスク、または外部磁気媒体などの形態である。制御プログラムは、装置例を管理するソフトウェアとして実現される。

【００６１】本発明はハードウェア、ソフトウェア、またはハードウェアとソフトウェアの組み合わせにより実現される。本発明は１コンピュータ・システム内で集中形式で実現されたり、異なる要素が複数の相互接続されるコンピュータ・システムに渡って広がる分散形式でも実現され得る。任意の種類のコンピュータ・システム、またはここで述べた方法を実現するように適応化された他の装置が使用可能である。ハードウェア及びソフトウェアの典型的な組み合わせは、コンピュータ・プログラムを有する汎用コンピュータ・システムであり、そこではコンピュータ・プログラムがロードされ、実行されると、コンピュータ・プログラムがコンピュータ・システムを制御して、ここで述べた方法を実行する。本発明はまた、コンピュータ・プログラム製品内に埋め込まれてもよい。この場合、コンピュータ・プログラム製品がここで述べた方法の実現を可能にする全てのフィーチャを含み、コンピュータ・システムにロードされて、これらの方法を実行する。

【００６２】ここでは、コンピュータ・プログラム手段またはコンピュータ・プログラムは、任意の言語、コードまたは表記法による、命令セットの任意の表現を意味し、ここで命令セットは情報処理能力を有するシステムに、特定の機能を直接的に実行させるか、或いは、１）別の言語、コードまたは表記法への変換、及び２）異なる材料形態での複製のいずれかまたは両方の後で、実行させる。

【００６３】前述の説明は、本発明の目的及び実施例の幾つかの概要を述べたものである。本発明の概念は、多くのアプリケーションに対して使用可能である。従って、前述の説明は特定の構成及び方法を述べたものであるが、本発明の趣旨及び概念は、他の構成及びアプリケーションにも適用可能である。例えば、データ・パケットについて言及したが、本発明は非データ・パケットにも同様に適用可能である。当業者であれば、本発明の趣旨及び範囲から逸れることなく、ここで開示された実施例の他の変更が可能であることが明らかであろう。従って、ここで述べた実施例は、単に本発明の傑出するフィーチャ及びアプリケーションの幾つかを表しただけであり、開示された本発明を異なる態様で適用することにより、または本発明を当業者に既知の方法で変更することにより、他の有益な利点が実現されることであろう。従って、ここで述べた実施例は１例として提供されただけで、本発明を制限するものではないことを、再度述べておく。

【００６４】まとめとして、本発明の構成に関して以下の事項を開示する。

【００６５】（１）データ・パケットを分類する方法であって、分類木のルート・ノードにおいて、データ・パケットを受信するステップと、前記分類木の第１の木レベルの第１の子が、前記第１の子のノード基準を満足することを示すまで、前記第１の木レベルのそれぞれの子に前記データ・パケットを連続的に受け渡すステップと、前記第１の子が前記データ・パケットを適合パケットに形成するステップと、続く次のレベルにおける次の木レベルの第１の子が、前記次の木レベルの前記第１の子のノード基準を満足しなくなるまで、前記次の木レベルに対して、受け渡し及び形成ステップを繰り返すステップとを含む、方法。

（２）前記受け渡すステップが、ステータス指示を返却するコード・セットを実行するステップを含む、前記（１）記載の方法。

（３）前記形成するステップが、前記第１の子が引き続き実行されるコード・セットを指定するステップを含む、前記（１）記載の方法。

（４）前記指定するステップが、満足に続いて実行されるコード・セットを指定するステップを含む、前記（３）記載の方法。

（５）前記分類木の少なくとも１レベルに、少なくとも１ノードを動的に追加するステップを含む、前記（１）記載の方法。

（６）前記少なくとも１つの新たな子ノードがリアル・オーディオ・ノードである、前記（５）記載の方法。

（７）パケットを分類する方法であって、前記パケットに対して進行中のパケット分類プロセスを延期するステップと、前記分類において使用される外部情報を獲得するステップとを含む、方法。

（８）前記獲得するステップが、前記外部情報により、

分類木内のノードのノード基準を増補するステップを含む、前記（７）記載の方法。
（９）前記外部情報が前記パケットの発信元の識別を含む、前記（８）記載の方法。
（１０）前記外部情報が前記パケットの発信元の認証を含む、前記（８）記載の方法。
（１１）分類プロセスが拡張可能な分類子プロセスである、前記（７）記載の方法。
（１２）前記適合パケットを構文解析し、関連情報を生成するステップを含む、前記（１）記載の方法。
（１３）前記適合パケットを変換パケットに変換するステップを含む、前記（１）記載の方法。
（１４）前記パケットを満足を示す最後の第１の子に関連付けるステップを含む、前記（１）記載の方法。
（１５）前記最後の第１の子に従い、コード・セットを実行するステップを含む、前記（１４）記載の方法。
（１６）前記データ・パケットの処置を決定するステップを含む、前記（１）記載の方法。
（１７）子ノードにおいて受信されたパケットの処置を決定する方法であって、前記パケット及び該パケットの第１の処置を外部プロセスに渡すステップと、前記外部プロセスがプロセス特定の手段を用いて、パケット処置を増補するステップと、増補パケット及び増補処置を前記子ノードに戻すステップとを含む、方法。
（１８）前記パケットに対して進行中の処置プロセスを延期するステップを含む、前記（１７）記載の方法。
（１９）前記増補処置が前記パケットの発信元の識別を含む、前記（１８）記載の方法。
（２０）前記増補処置が前記パケットの発信元の認証を含む、前記（１８）記載の方法。
（２１）前記処置がポリシ施行のために使用される、前記（１８）記載の方法。
（２２）分類プロセスをファイアウォールとして使用するステップを含む、前記（１６）記載の方法。
（２３）分類プロセスをアプリケーション・レベルの分類のために使用する、前記（１）記載の方法。
（２４）分類プロセスをポリシ施行のために使用する、前記（２３）記載の方法。
（２５）分類プロセスを速度制限のために使用する、前記（２３）記載の方法。
（２６）分類プロセスを負荷平衡化のために使用する、前記（２３）記載の方法。
（２７）分類プロセスをトラフィック形成のために使用する、前記（１）記載の方法。
（２８）データ・パケットを分類する装置であって、物理ネットワークからデータ・パケットを受信し、該データ・パケットを分類木のルート・ノードに渡し、また逆に、前記ルート・ノードからデータ・パケットを受信し、該データ・パケットを前記物理ネットワークに送信するネットワーク・インタフェース装置と、分類木の次の木レベルの第１の子ノードが、該第１の子ノードのノード基準を満足することを示すまで、前記次の木レベルにおいて、子ノードから子ノードへとパケットを連続的に受け渡し、続く次のレベルの第１の子ノードが、前記続く次のレベルの前記第１の子ノードのノード基準を満足しなくなるまで、データ・パケットを適合パケットに形成するパケット・モジュールとを含む、装置。
（２９）前記装置の一部がアクセラレータ・チップとして実現される、前記（２８）記載の装置。
（３０）前記装置がアプリケーション・レベルの分類に使用される、前記（２８）記載の装置。
（３１）前記装置がファイアウォールとして使用される、前記（２８）記載の装置。
（３２）前記装置がボーダ・サーバとして使用される、前記（２８）記載の装置。
（３３）前記ステータス指示がｐｍ＿ｔタイプである、前記（２）記載の方法。
（３４）データ・パケットの分類を行うコンピュータ読取り可能プログラム・コード手段を有するコンピュータ読取り可能媒体を含む製品であって、前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに前記（１）のステップを実行するように指示するコンピュータ読取り可能プログラム・コード手段を含む製品。
（３５）前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに分類木の少なくとも１レベルに少なくとも１ノードを動的に追加するように指示する、前記（３４）記載の製品。
（３６）データ・パケットの分類を行うコンピュータ読取り可能プログラム・コード手段を有するコンピュータ読取り可能媒体を含む製品であって、前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに前記（８）のステップを実行するように指示するコンピュータ読取り可能プログラム・コード手段を含む製品。
（３７）パケットの処置の決定を行うコンピュータ読取り可能プログラム・コード手段を有するコンピュータ読取り可能媒体を含む製品であって、前記コンピュータ読取り可能プログラム・コード手段が、コンピュータに前記（１８）のステップを実行するように指示するコンピュータ読取り可能プログラム・コード手段を含む製品。
（３８）データ・パケットを分類する装置であって、分類木のルート・ノードにおいて、データ・パケットを受信する手段と、前記分類木の第１の木レベルの第１の子ノードが、前記第１の子ノードのノード基準を満足することを示すまで、前記第１の木レベルのそれぞれの子に前記データ・パケットを連続的に受け渡す手段と、前記第１の子が前記データ・パケットを適合パケットに形成するステップと、続く次のレベルにおける次の木レベルの第１の子ノードが、前記続く次のレベルの前記第１の子ノードのノード基準を満足しなくなるまで、前記次の木レベルに対して、受け渡し及び形成ステップを繰り返

す手段とを含む、装置。

（３９）子ノードにおいて受信されるパケットの処置を決定する装置であって、前記子ノードが存在する、制御プログラムの割込みコンテキストと、前記制御プログラムの割込みコンテキストの範囲外の外部プロセスと、前記パケット及び該パケットの第１の処置を前記外部プロセスに渡し、前記外部プロセスにプロセス特定手段の使用によりパケット処置を増補させ、増補パケットを増補処置と一緒に子ノードに返却させる手段とを含み、前記割込みコンテキストが、前記増補パケット及び前記増補処置を前記外部プロセスから受信する手段を含む、装置。

【図面の簡単な説明】

【図１】プロトコル層とＴＣＰ／ＩＰプロトコル・スタックとの関係を示す図である。

【図２】ウェブ・サーバに送信される前に、ＨＴＴＰ要求がカプセル化されるステージを示す図である。

【図３】入来ＨＴＴＰ要求に対して分類が行われる様子を示す図である。

【図４】本発明に従い分類木内のモジュールの編成方法の１例を示す図である。

【図５】本発明に従いパケットを分類するパケット分類及び逆多重化プロセスの１例を示す図である。

【図６】本発明に従いパケット処置を決定するステップの１例を示す図である。

【図７】本発明に従うｐｍ＿ｔ戻りコードの１例を示す図である。

【図８】本発明に従うアプリケーション依存ノードの１例を示す図である。

【図９】本発明に従うｐｐ＿ｔ戻りコードの１例を示す＊図である。

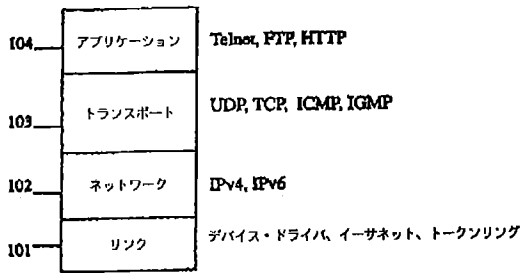
【図１０】本発明に従うpaction_t戻りコードの１例を示す図である。

【図１１】本発明に従う装置の１例を示す図である。

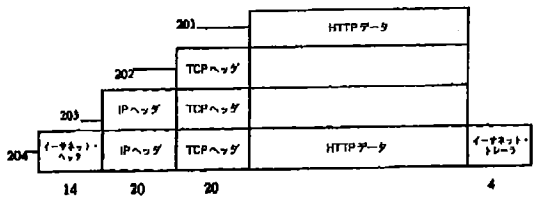
【符号の説明】

１０１、３００　リンク層
１０２、３１０　ネットワーク層
１０３、３２０　トランスポート層
１０４、３３０　アプリケーション層
３０１　入来ＨＴＴＰ要求
３０２　イーサネット・ドライバ
３１２　ＩＰｖ４
３２３　ＴＣＰ
３３２　ＨＴＴＰサーバ
５０２　ルート・ノード
５０３　ＩＰｖ４
５０４　ＩＰｖ６
５０６　ＵＤＰ
５０７　ＨＴＴＰ
５０８　ＴＣＰ
５２１　第１レベル
６０３　パケット・マッチング機能（ｐｍ）
６２１　パケット・アクション機能
７００　ｐｍ＿ｔタイプ戻りコード値
８３１　Ｈ．３２３
８３２　リアル・オーディオ
８３３　ＦＴＰ
９００　ｐｐ＿ｔタイプ戻りコード値
１１０１　ネットワーク・インタフェース装置
１１０３　パケット・モジュール

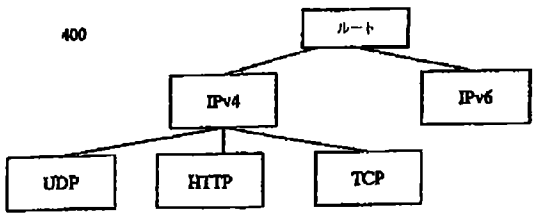
【図１】



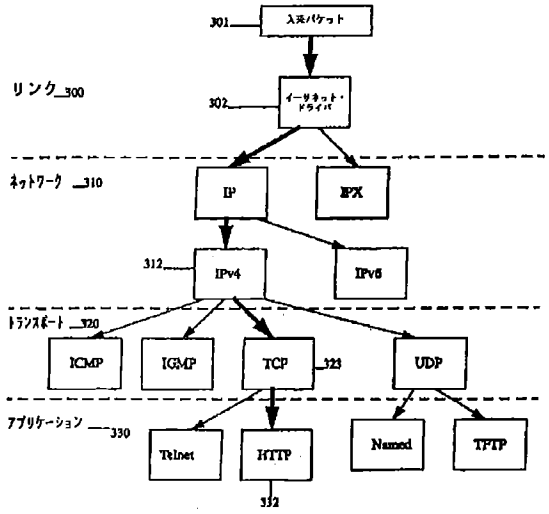
【図２】



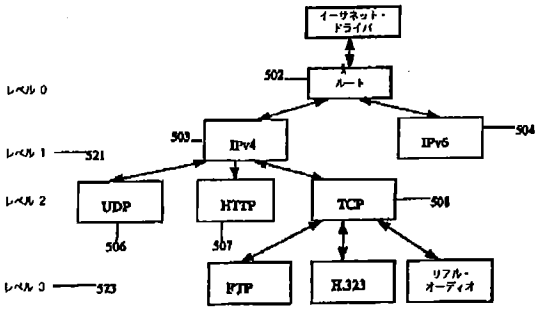
【図４】



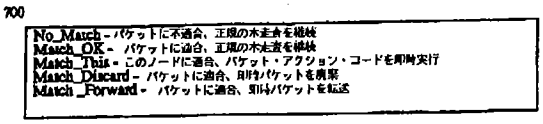
【図３】



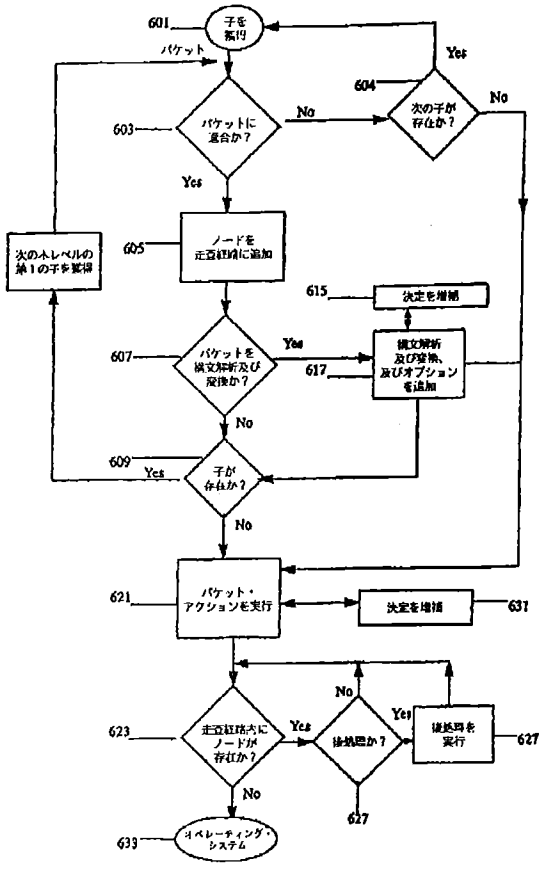
【図５】



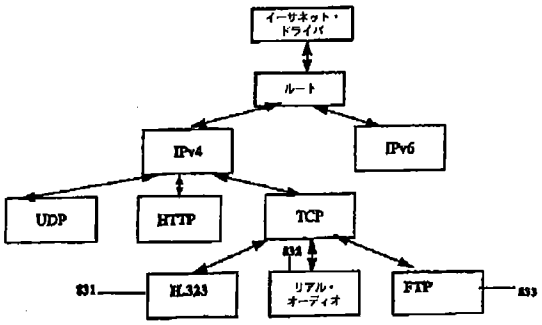
【図７】



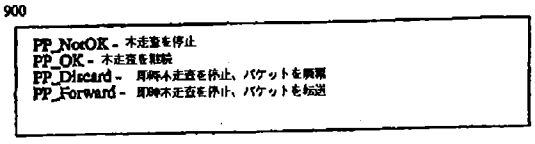
【図６】



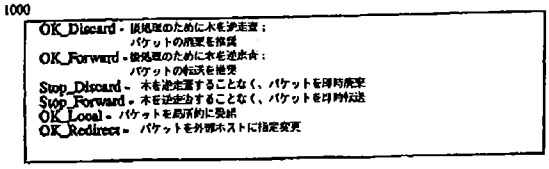
【図８】



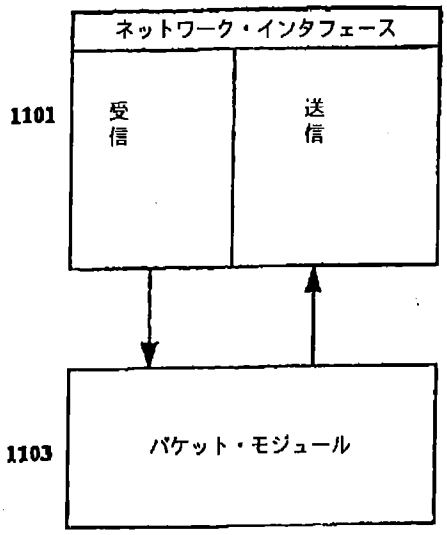
【図９】



【図１０】



【図１１】



フロントページの続き

(72)発明者　ダグラス・リー・シャルス
アメリカ合衆国10549、ニューヨーク州サウス・セーラム、エリンウッド・ロード　136
(72)発明者　スリニバサン・セシャン
アメリカ合衆国10583、ニューヨーク州スカースデール、アパートメント　2、ガース・ロード　142
(72)発明者　ミリアン・ゾハー
アメリカ合衆国10977、ニューヨーク州ニュー・ヘンプステッド、ペニントン・ウェイ　31
Ｆターム(参考)　5K030 GA01 HA08 HB18 JA05
5K034 AA01 BB06 HH01 HH02 HH06
HH63